AFRL-HE-WP-TP-2006-0052

AIR FORCE RESEARCH LABORATORY

Visual Performance-based Image Enhancement Methodology: An Investigation of Contrast Enhancement Algorithms

Kelly E. Neriani
Travis J. Herbranson
George A. Reis
Alan R. Pinkus

Human Effectiveness Directorate
Warfighter Interface Division
Wright-Patterson AFB OH 45433-7022

Charles D. Goodyear

General Dynamics
5200 Springfield St., Suite 200
Dayton OH 45431

March 2006

Human Effectiveness Directorate
Warfighter Interface Division
Wright-Patterson AFB OH 45433

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
March 2006	Technical Paper	

4. TITLE AND SUBTITLE
Visual Performance-based Image Enhancement Methodology: An Investigation of Contrast Enhancement Algorithms

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
**Kelly E. Neriani, **Travis J. Herbranson, **George Reis,
**George Reis, *Charles D. Goodyear

5d. PROJECT NUMBER
7184

5e. TASK NUMBER
11

5f. WORK UNIT NUMBER
27

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES)
*General Dynamics
5200 Springfield St., Suite 200
Dayton OH 45431

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
**Air Force Materiel Command
Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Interface Division
Wright-Patterson AFB OH 45433-7022

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/HECV

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-HE-WP-TP-2006-0052

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
This will be published in the Proceedings of the SPIE Defense and Security Symposium. The clearance number is AFRL/WS-06-0712, cleared 15 March 2006.

14. ABSTRACT
While vast numbers of image enhancing algorithms have already been developed, the majority of these algorithms have not been assessed in terms of their visual performance-enhancing effects using militarily relevant scenarios. The goal of this research was to apply a visual performance-based assessment methodology to assess six algorithms that were specifically designed to enhance the contrast of digital images. The image enhancing algorithms used in this study included three different histogram equalization algorithms, the Autolevels function, the Recursive Rational Filter technique described in Marsi, Ramponi, and Carrato and the multiscale Retinex algorithm described in Rahman, Jobson and Woodell. The methodology used in the assessment has been developed to acquire objective human visual performance data as a means of evaluating the contrast enhancement algorithms. The basic approach is to use standard objective performance metrics, such as response time and error rate, to compare algorithm enhanced images versus two baseline conditions, original non-enhanced images and contrast-degraded images. Observers completed a visual search task using a spatial-forced-choice paradigm. Observers searched images for a target (a military vehicle) hidden among foliage and then indicated in which quadrant of the screen the target was located.

15. SUBJECT TERMS
Visual performance, methodology

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Kelly E. Nerinai
UNC	UNC	UNC	SAR	14	19b. TELEPHONE NUMBER *(include area code)* (937) 255-5128

Visual performance-based image enhancement methodology: an investigation of contrast enhancement algorithms

Kelly E. Neriani*[ab], Travis J. Herbranson[bc], George A. Reis[b], Alan R. Pinkus[b], and Charles D. Goodyear[d]
[a]Consortium Research Fellows Program, 2511 Jefferson Davis Highway, Arlington, VA, USA 22202-3926
[b]Air Force Research Laboratory, Battlespace Visualization Branch, 2255 H Street, Wright-Patterson AFB, OH 45433-7022
[c]Air Force Institute of Technology, Department of Operational Sciences, 2950 Hobson Way, Wright Patterson AFB, OH 45433-7765
[d]General Dynamics-Advanced Information Engineering Systems, 5200 Springfield Pike, Suite 200, Dayton, OH 45431-1289

ABSTRACT

While vast numbers of image enhancing algorithms have already been developed, the majority of these algorithms have not been assessed in terms of their visual performance-enhancing effects using militarily relevant scenarios. The goal of this research was to apply a visual performance-based assessment methodology to assess six algorithms that were specifically designed to enhance the contrast of digital images. The image enhancing algorithms used in this study included three different histogram equalization algorithms, the Autolevels function, the Recursive Rational Filter technique described in Marsi, Ramponi, and Carrato[1] and the multiscale Retinex algorithm described in Rahman, Jobson and Woodell[2]. The methodology used in the assessment has been developed to acquire objective human visual performance data as a means of evaluating the contrast enhancement algorithms. The basic approach is to use standard objective performance metrics, such as response time and error rate, to compare algorithm enhanced images versus two baseline conditions, original non-enhanced images and contrast-degraded images. Observers completed a visual search task using a spatial-forced-choice paradigm. Observers searched images for a target (a military vehicle) hidden among foliage and then indicated in which quadrant of the screen the target was located. Response time and percent correct were measured for each observer. Results of the study and future directions are discussed.

1. INTRODUCTION

There are several different techniques to assess the relative improvement in image quality when an image enhancing algorithm has been applied to a digital image. The testing of enhancing effects often consists of subjective quality assessments or measures of the ability of an automatic target detection program to find a target before and after an image has been enhanced. It is rare to find studies that focus on the human ability to detect a target in an enhanced image using scenarios that are relevant for the particular application for which the enhancement is intended.

For instance, a recent study by Sale, Schultz, and Szczerba[3] used a Bayesian super-resolution enhancement algorithm on an infrared digital video to determine the effectiveness of the enhancement for military purposes. The results were obtained using a before/after subjective assessment. The images presented did appear to have substantially finer detail; however, one cannot say whether this increase in detail would result in better target detection. In another study, by Rahman, Jobson and Woodell[2], a variation of the Retinex algorithm (the MSRCR) was studied. The images before and after processing with the MSRCR algorithm were compared and assessed in terms of image quality using subjective methods. The images were also assessed using an automated method of determining image quality, by assigning regions of the unprocessed and processed images to categories of excellent, good, or poor, based on global and regional brightness and contrast measures. This automatic method did result in a numerical assessment of quality, but it is unknown how this value would relate to human performance.

While a particular algorithm may make an image appear substantially better after enhancement, there is no indication as to whether this improvement is significant enough to improve human visual performance. Therefore, Neriani, Herbranson, Pinkus, Task and Task[4] developed a methodology that used a visual search task to determine the effect, if any, that image enhancing algorithms have on improving human visual performance. Since the aim of the research was to improve human visual performance, an improvement in image quality was quantified as a decrease in response time when observers perform a visual search task. Specifically, the research was designed to measure performance on the militarily relevant task of searching for a target hidden among foliage. The methodology used gave a precise and useful estimate as to how much (if at all) the observers' performance improved when the target images were enhanced with each of the three Retinex algorithms that were studied.

Using the methodology developed in Neriani et al.[4], the research discussed in this paper is focused on the assessment of six different algorithms designed to enhance the contrast of digital images. These algorithms are described in some detail in the next section.

2. ALGORITHMS

2.1. Global Histogram Equalization (HE)

The global histogram equalization process is a contrast enhancing technique that reassigns the brightness values of the pixels in an image based on the image's histogram. The individual pixels retain their brightness order (they remain brighter or darker than other pixels) but the values are shifted, so that an equal number of pixels have each possible brightness value. In many images this will spread out the values in parts of the image where different regions meet, showing detail in areas with a high brightness gradient.

The process is the following:

For each brightness level j in the original image (and its histogram), the new assigned value k is calculated as:

$$k = \sum_{i=0}^{j} N_i / T \tag{1}$$

where the sum counts the number of pixels in the image (by integrating the image histogram) with a brightness less than or equal to j, and T is the total number of pixels in the image (area under the curve in the histogram).

2.2. Autolevels

Autolevels is a commonly used image enhancing algorithm that adjusts input images that suffer from low dynamic range. Low dynamic range can occur when the image is too bright and in this case the image histogram will have many values in the high end and very few in the low end. Likewise, low dynamic range can occur when the image is too dark. In this case, the histogram will be skewed to have more values at the low end of the range and very few in the high end.

Autolevels will automatically detect and fix this kind of imbalance. It scans through the levels of intensity within the image and chooses a level that should be regarded as black (low intensity) and another that should be regarded as white (high intensity). It then stretches the levels in the image so that all the intensities present lie between the black and the white points. This results in an image with a good span of intensities.

To help mitigate the effect of outliers - small numbers of pixels at extreme values of intensity - the tails of the image histogram are clipped using a pre-defined parameter. The parameter used clips the tails as a percentage of the total number of pixels in the image. In the images used in the current study, the

Autolevels algorithm was applied with a clipping percentage of 0.5, which means that the bottom and top 0.5% of pixels in the histogram will be ignored when determining the black and white points.

2.3. Multiscale Retinex (MSR)

The MSR is based on the concept of the Retinex as developed by Edwin Land[5] as a model of the lightness and color perception of human vision. The basic idea of the Retinex is to predict the sensory response of lightness. The model in Land and McCann[6] describes four steps for every iteration of a Retinex calculation. The model uses operators that sum, difference, and rectify the values in an image to obtain spatial interactions between the pixels.

One of the important concepts behind the Retinex model's computation of lightness for any given pixel is the comparison of that pixel's value with the values of the other pixels in the image. The MSR uses a center/surround design based on the receptive fields of cells found in the visual system of primates (starting in the retina as retinal ganglion cells).

Equation 2 describes the multiscale Retinex:

$$R_i(x_1, x_2) = \sum_{k=1}^{K} W_k \{\log I_i(x_1, x_2) - \log[F_k(x_1, x_2) * I_i(x_1, x_2)]\} \quad i = 1, ..., N, \qquad (2)$$

where index i refers to the i^{th} spectral band, (x_1, x_2) is the pixel location, and * represents the convolution operator. N is the number of spectral bands (N is three for our images, which are R, G, B), I is the input image and R is the output image after processing. F_k is the k^{th} (Gaussian) surround function, W_k is the weighting associated with F_k, and K is the number of surround functions (or scales). F_k is defined as:

$$F_k(x_1, x_2) = \kappa \exp[-(x_1^2 + x_2^2)/\sigma_k^2], \qquad (3)$$

where σ_k are the standard deviations of the Gaussian surrounds. Two different surround functions were used for this research:

1. $\sigma_k = 0.5$, with $W_k = 1.0$
2. $\sigma_k = 15.0$, with $W_k = 1.0$

The MSR output is normalized by $\kappa = 1/[\sum_{x_1} \sum_{x_2} F(x_1, x_2)]$. (4)

2.4. Partially Overlapped Sub-block Histogram Equalization (POSHE)

The POSHE algorithm is an advanced histogram equalization algorithm. It realizes the high contrast enhancing effect associated with a local histogram equalization process, while maintaining the lower computational complexity of a global histogram equalization process. The algorithm is described briefly in this section. For equations and more details, see Kim, Kim and Hwang[7].

In the case of global histogram equalization, histogram equalization is performed over the entire image at once, using the method described above. In the local histogram equalization process, also referred to as block-overlapped histogram equalization, a sub-block of the image is defined and the histogram of the sub-block is collected. Then, histogram equalization is performed on the center pixel of the sub-block by using the cumulative distribution function (cdf) of that sub-block. Next, the sub-block is moved over by one pixel and the process is repeated until the end of the input image is reached.

To achieve the effect of a local histogram equalization process while maintaining lower computational complexity, partially-overlapped sub-block histogram equalization is performed.

First, one has to define an M x N sized output image array for an M x N sized input image and set all the starting values in the array to zero. The next step is to assign an m x n sized sub-block. The sub-block size is equal to the quotient of the input image size divided by a multiple of two. We divided by the value of 4 in this experiment. The sub-block origin is assigned by using the input image origin. Next, local histogram equalization is done on the current sub-block and results are accumulated in the output image array.

Once the current sub-block has been processed, the horizontal coordinate of the sub-block origin is increased by a horizontal step size and another local histogram equalization is performed on the current sub-block. When the horizontal coordinate hits the edge of the image (equals the horizontal input image size), the vertical coordinate of the sub-block origin is increased by a vertical step size, and then the horizontal POSHE is done. This process is repeated for the entire image. The last step is to divide each pixel value in the output image array by its sub-block histogram equalization frequency. We used a horizontal step size of 8 and a vertical step size of 8.

One of the issues that exists with POSHE are the blocking effects that occur at the boundaries of the sub-blocks. As suggested in Kim et al.[7], if there is a blocking effect at the sub-block boundaries, apply a blocking effect reduction filter (BERF). The BERF suggested in Kim et al.[7] was applied to the test images used in the current study.

2.5. Block-based Binomial Filtering Histogram Equalization (BBFHE)

The Block-based binomial filtering histogram equalization (BBFHE) as described in Lamberti, Montrucchio, and Sanna[8], has the same theoretical basis as the POSHE algorithm, and as such, has a similar implementation. The authors state that the BBFHE is improved with respect to the POSHE algorithm in terms of computational complexity and similar to the POSHE with respect to visual quality. The difference in the implementation of BBFHE is in the filter that is used. POSHE uses a low pass filter over the histograms of the sub-blocks. BBFHE uses a binomial filter, which the authors state is improved in terms of speed and easily able to be implemented in hardware.

A binomial filter with p= 4, corresponding to a 5 x 5 filter mask, is used in this research, where p is the order of the filter. The steps of the BBFHE algorithm are as follows: first, the input image is divided into sub-blocks and histograms are calculated for each sub-block. Then, the decomposed two-dimensional binomial filter is applied to each sub-block histogram and its neighbors. Finally, the histograms that result are used to perform the histogram equalization on each sub-block. To correct for blocking effects at the sub-block boundaries, the binomial histogram equalization was performed 15 times on partially shifted input images (shifted by 4 pixels horizontally and 3 pixels vertically) and the results were averaged to get the final image. For more details, see the original source, Lamberti et al.[8].

2.6. Recursive Rational Filter (RRF)

The recursive rational filter approach to contrast enhancement is summarized in detail in Marsi et al.[1]. The new method discussed is based heavily on the Retinex approach, described in some detail above. The difference between the RRF method and a standard Retinex method lies in the corrected estimation of the illumination component. The algorithm is mainly based on rational filters, working in a recursive manner, and is low in computational intensity, making it suitable for a real-time application.

The goal for the estimation of illumination is to create an edge preserving low-pass filter with a quite narrow band. However, this implies a wide impulse response of the filter, causing the algorithm to become highly computationally complex. Therefore, the RRF method uses the spatial recursivity of the operator to get narrow bands with only a few input taps.

To calculate the illumination estimate, $\tilde{L}(x,y)$, the following equations are used:

$$a = \alpha \frac{S}{S+1} \tag{5}$$

where

$$S = \frac{H}{\left(\log \frac{1 + x(n-1)}{1 + x(n+1)}\right)^2 + \delta} \tag{6}$$

where α is the maximum value that the a coefficient can assume and consequently is correlated to the minimal bandwidth of the filter, S is an appropriate sensor (that can detect edges): when $x(n-1)$ and $x(n+1)$ have similar values, S will become large and when $x(n-1)$ and $x(n+1)$ both have very different values, S decreases (detects sharp edge). H is a parameter used to set the intensity of the sensor response and δ is a small value used in order to avoid the denominator becoming zero. In this experiment, α = 0.75, a = 0.3, H = 0.01 and δ = 0.00001.

This algorithm can be extended in two directions, horizontal and vertical. S_0 and S_v are used respectively to estimate the horizontal and vertical transitions in the signal, where m and n are the spatial coordinates of the signal.

$$S_o = \frac{H}{\left(\log \frac{1 + x(n, m-1)}{1 + x(n, m+1)}\right)^2 + \delta} \tag{7}$$

$$S_v = \frac{H}{\left(\log \frac{1 + x(n-1, m)}{1 + x(n+1, m)}\right)^2 + \delta} \tag{8}$$

The original input image is divided, pixel by pixel, by this illumination estimate (calculated in the equations above), called $\tilde{L}(x,y)$ to get an estimate of the reflectance. The log is taken of these values and then they are processed through a sigmoid-like function described using equation 9:

$$R = 2K\left(\frac{1}{1 + \exp(-cR')} - \frac{1}{2}\right) \tag{9}$$

where

$$\text{if } |R'| < d, \text{ then } c = h_1 + \frac{h_2 - h_1}{d}|R'|, \quad \text{else} \quad c = h_2. \tag{10}$$

K defines the level of emphasis, c controls the gradient of the sigmoid, and h_1, h_2, and d are responsible, respectively, for controlling the slope of the sigmoid in the origin, its curvature, and the amplitude of the "central dead zone". In this experiment, we used the values of K = 1, h_1 = 0.3, h_2 = 2.0, and d = 0.4

Next, an exponential function carries out the inverse of the exponential function used above. Before the illumination estimate and reflectance estimate are combined, the estimated illumination, $\tilde{L}(x,y)$, is processed through a non-linear function similar to a gamma correction, using the following equation:

$$\Gamma(\tilde{L}) = 255\left(\frac{\tilde{L}}{255}\right)^{a^{\frac{\tilde{L}}{255+a}}} \quad (11)$$

where a = 0.3. Finally, a histogram stretch is applied to these values and the resulting illumination estimate, $\tilde{L}(x,y)$, and the reflectance estimate, $\tilde{R}(x,y)$, are multiplied together to get the values for the final image.

3. METHODS

The goal of this research was to use a methodology that incorporates a measure of human visual performance in the assessment of the effectiveness of several contrast enhancing algorithms. A standard psychophysical task (spatial-forced-choice visual search task) was employed to measure human visual performance.

3.1. Observers

Seven observers including five males and two females participated in the experiment. The observers ranged in age from 21 to 50 years. All had normal color vision and normal or corrected to normal visual acuity. All of the observers completed at least three practice sessions on the task before the data reported were collected.

3.2. Stimuli

In the experiment, each observer viewed a total of 1696 grayscale images. Of these, 1536 images consisted of a target located in a scene of trees and grass. The target was a model of a Renault R39 reconnaissance tank (see Figure 1), placed on an artificial terrain board. For the rest of the 1696 test images, the observers viewed 160 images of an artificial terrain board scene of trees and grass with no target as catch trials. These catch trials were used to help reduce guessing by the observer since they were told there were some trials with no target.

The images were taken with the artificial terrain board rotated at 16 different angles (spanning from 0° - 337.5°) and at each rotation angle, the camera was positioned on the tripod at three different tilts (tilted to the left, in the center, and tilted to the right). If a target was present, the target was placed so that it clearly fell into one of the four quadrants of the screen (see Figure 4). Additionally, to control for the effect of density of foliage, the targets were always placed in an area of medium masking, which was defined as the target being located along a tree line or directly adjacent to a clump of bushes.

All 1696 images were presented using one of eight different algorithm conditions. These conditions were a non-degraded condition with no algorithm, a contrast-degraded condition with no algorithm, and six conditions corresponding to contrast-degraded images processed by each of the six contrast-enhancing algorithms described above. The images for the non-degraded condition with no algorithm were taken with the digital camera set to have an exposure of 1 second, with the camera one meter away from the terrain board, illuminated with flood lights (see Figure 2). The images for the contrast-degraded condition were taken at the same time as the non-degraded condition with no algorithm images and were the same except captured with an exposure of 125 milliseconds. The 1536 images with a target present included one trial for each combination of 16 rotation angles x 3 camera tilts x 4 quadrants x 8 algorithm conditions.

Figure 5 shows the eight different algorithm conditions applied to one scene (be aware of differences between images displayed on the CRT and images displayed on paper). The arrow in the top-left image highlights the target which is the placed in the same location for each image in Figure 5.

The images were taken using a Nikon COOLPIX E8800 8.0 megapixel digital camera with a resolution of 1280 x 960 pixels. Before the images were used in the experiment, both the enhanced and non-enhanced

images were resized to a resolution of 1155 x 866 pixels. This was necessary due to the constraints of the program running the experiment.

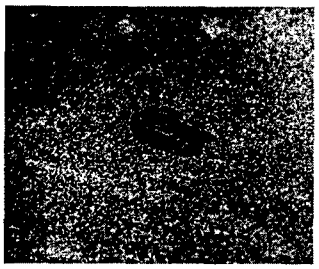

Figure 1. Target used in the experiment.

Figure 2. Terrain board used for scenery in the experiments. The image also shows the camera angle and equipment setup used to capture the images.

3.3. Procedure

The observers viewed the images on a 21-inch ViewSonic G220fb color monitor driven by a Diamond Savage4 video card in a 700 MHz Pentium III Micron Computer placed on a desktop. The observers were seated in a comfortable chair in a dimly lit room during the experimental sessions and viewed the images from a distance of about 36 inches. Prior to the collection of data, the observers were instructed to perform a visual search on each image shown to locate a target and that once they located the target, they would be required to indicate which quadrant the target was located in. Additionally, the observers were told that some of the images shown would not have a target. Before each trial started, the observers fixated on a blank green screen with a black fixation cross in the center of it (see Figure 3). Observers pressed the spacebar on the keyboard to initiate a trial whenever they were ready. Once the spacebar was pressed, the image was displayed. The observer pressed the spacebar again as soon as they determined whether the target was present or absent and the displayed image was replaced with an image indicating the assignment of the four quadrants (see Figure 4). Then, the observer pressed the number on the keyboard that corresponded to the quadrant the target was located in (1-4) or zero if there was no target in the image. Each trial had a time limit of 20 seconds. If the observer did not press the spacebar in 20 seconds or less once the test image was displayed, the test image was automatically removed and the quadrant screen was displayed. Observers were then forced to choose which quadrant the target was in or respond zero if they did not find the target. Percent correct and response time (measured from the first display of the image until the spacebar was pressed to indicate target presence or absence) were recorded for each trial.

Each observer had four 15-20 minute sessions on each of eight days. During each session, the observer viewed 48 images with a target present and five images with no target. The presentation order across all sessions for each observer was randomized with the constraint that each of the eight algorithm conditions was used for six trials within each session.

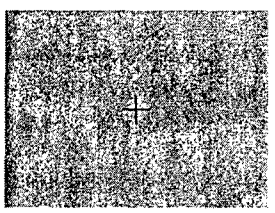

Figure 3. Fixation image to ready the observer for the start of a new trial.

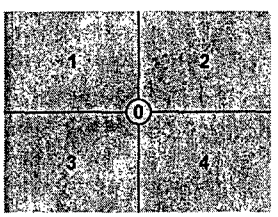

Figure 4. Image used to indicate quadrant assignment.

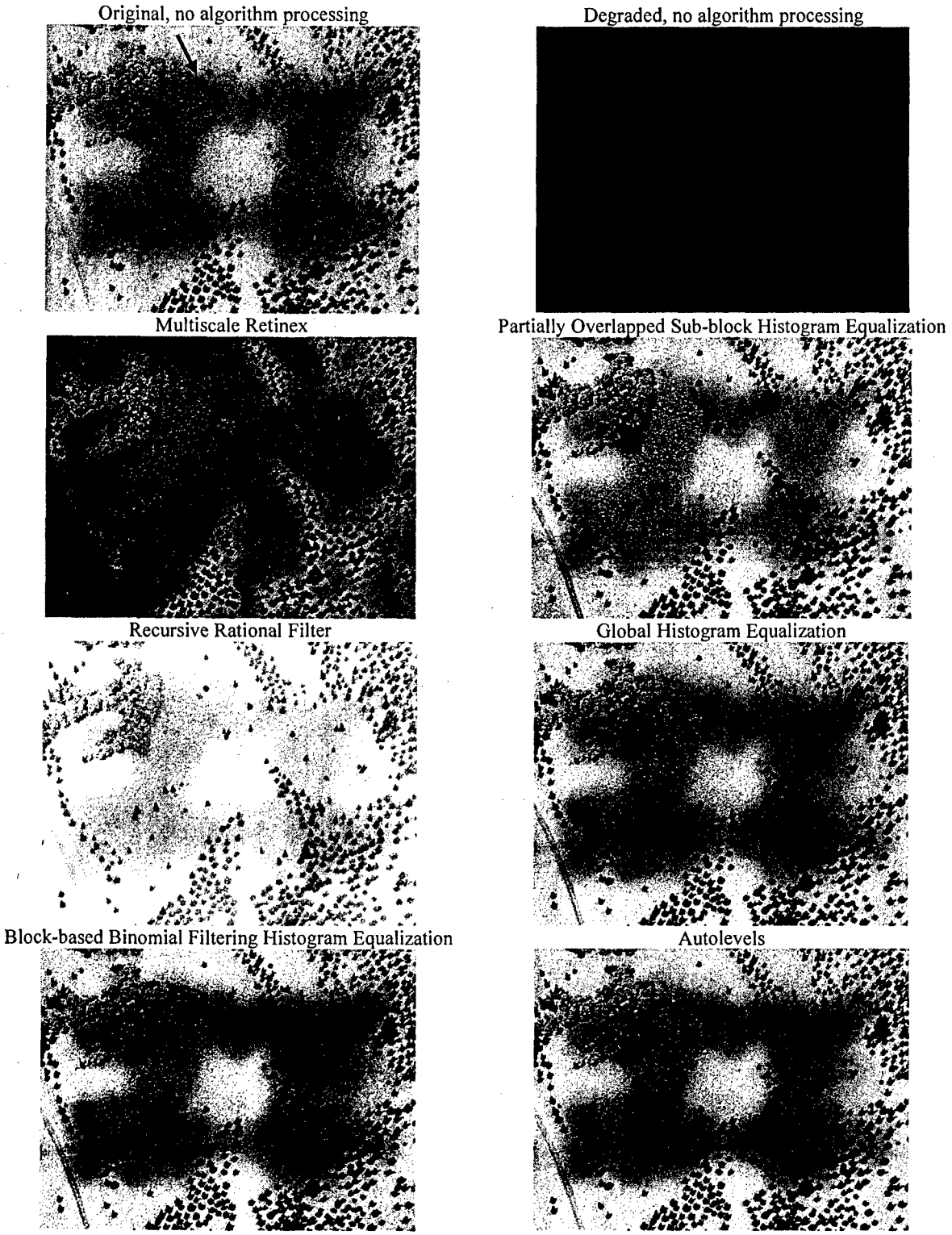

Figure 5. The eight different algorithm conditions. The arrow in the top left image highlights the target.

4. RESULTS

The dependent measures of response time and percent error were measured for each trial. Including all seven observers, there were 11,872 trials. Of these, there were 1120 catch trials. There were 163 catch trials (15%) in which an observer thought they saw a target. These trials were not used for any further analyses.

Across all observers there were 10,752 trials in which a target was present. Of these, there were 885 timed-out trials (8%) in which the observer failed to press the spacebar in 20 seconds or less, 99 trials (1%) where the observer decided there was no target (before 20 seconds), and 112 trials (1%) where the observer pushed the button for the wrong quadrant. These three categories of error added up to a total error of 10%.

Both incorrect responses and timed-out responses were combined into percent error. For the 211 incorrect trials in which observers said there was no target or identified the wrong quadrant as containing the target, there is no meaningful response time for finding a target. However, for the 885 timed-out trials, one can state that the response time for finding a target was greater than 20 seconds. For response time, a comparable measure of central tendency for each algorithm condition was determined within each observer using the following steps. The first step was to identify which of the 192 combinations of 16 rotation angles x 3 camera tilts x 4 quadrants contained a response time for each algorithm condition. That is, the combinations in which none of the 8 algorithm conditions had an incorrect trial. Then, for those combinations having a response time for each algorithm condition, the median response time across these combinations was determined for each algorithm condition.

The same procedure that was used to determine a comparable median response time for each observer and algorithm condition was used to determine a comparable median response time for each observer and rotation angle, each observer and camera tilt, and each observer and quadrant.

Both the median response time and percent error determined for each observer and algorithm condition were positively skewed, therefore, a log transformation was used for the analyses. Repeated measures analyses of variance were performed with the log of median response time and percent error as the dependent variables. The factor was algorithm condition. F-tests showed a significant difference among the algorithm conditions for the log of median reaction time $\{F(7,42) = 48.0, p = 0.0001\}$ and for the log of percent error $\{F(7,42) = 70.3, p = 0.0001\}$. Post hoc paired comparisons used the Least Significant Difference (LSD) procedure with a 0.01 per comparison error level.

The F-test showed a significant difference among the rotation angles for log of median response time $\{F(15,90) = 9.5, p = 0.0001\}$ and for the log of percent error $\{F(15,90) = 6.8, p = 0.0001\}$. The F-test did not show a significant difference among the camera tilts for the log of median response time $\{F(2,12) = 2.6, p = 0.1157\}$ or for the log of percent error $\{F(2,12) = 0.8, p = 0.4664\}$. The F-test showed a significant difference among the quadrants for the log of median response time $\{F(3,18) = 11.8, p = 0.0002\}$ and for the log of percent error $\{F(3,18) = 23.2, p = 0.0001\}$.

Figure 6 contains the results of the paired comparisons with algorithm conditions sorted by increasing response time. The whiskers represent the least significant difference value. This value is the mean difference between a pair of algorithm conditions that would have a p-value of 0.01. The upper panel shows the mean response time for each algorithm condition. The lower panel shows the percent error for each algorithm condition. No whiskers are shown for percent error of the Original and RRF algorithm conditions because the whiskers cut through the means. The Original and the RRF algorithm conditions were not significantly different from each other. Figure 7 contains results of the paired comparisons with rotation angles sorted by increasing response time. The lower panel of Figure 7 contains results of the paired comparisons of percent error. The whiskers represent the least significant difference value (as described above). Figure 8 contains results of the paired comparisons with camera tilts sorted by increasing response time. Figure 9 contains the results of the paired comparisons with quadrant sorted by increasing response time. The whiskers represent the least significant difference value (as described above).

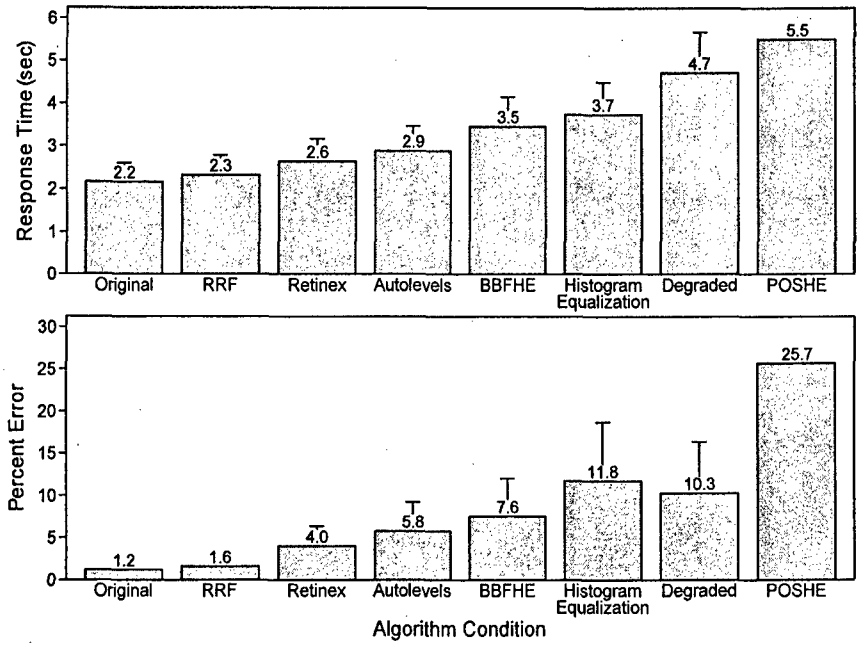

Figure 6. Mean response time and percent error (across observers) by algorithm condition, transformed back from log units used in the analyses.

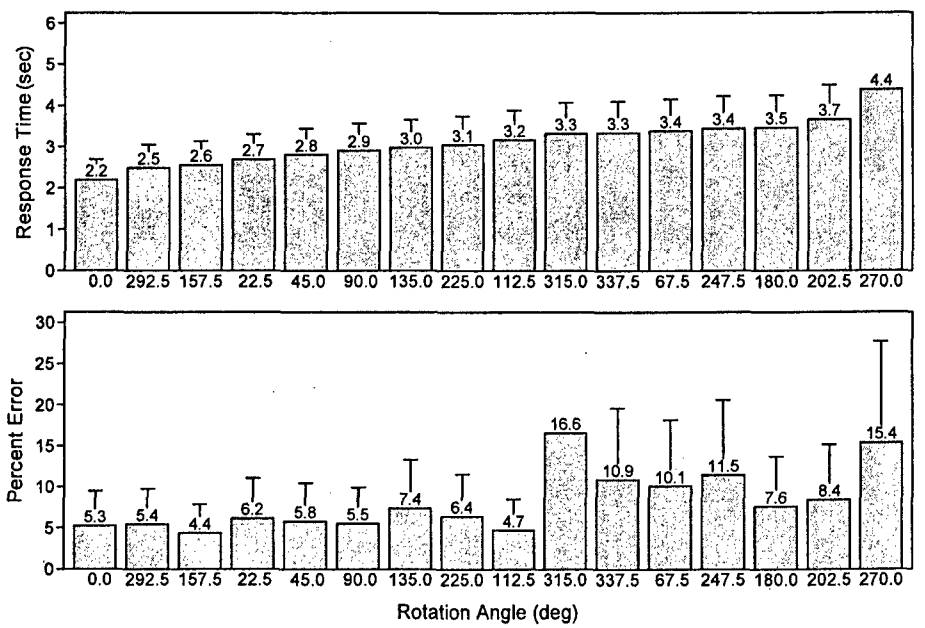

Figure 7. Mean response time and percent error (across observers) by rotation angle, transformed back from log units used in the analyses.

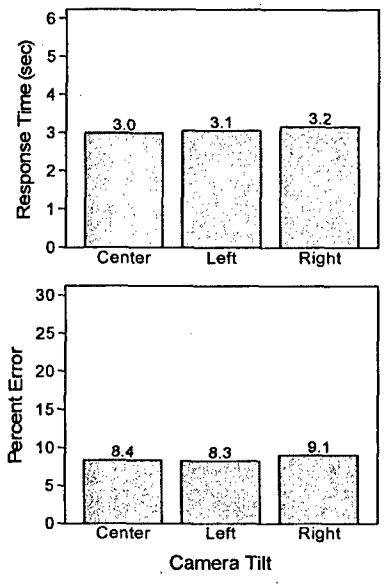

Figure 8. Mean response time and percent error (across observers) by camera tilt.

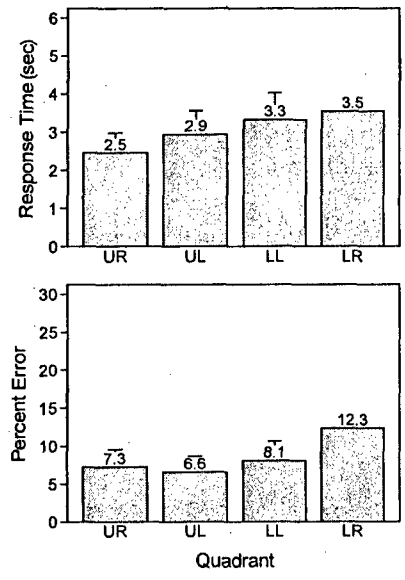

Figure 9. Mean response time and percent error (across observers) by quadrant.

5. DISSCUSSION / CONCLUSIONS

Figure 6 summarizes the results of the paired comparisons, sorted by increasing response time. As you can see, the RRF algorithm performed very well, with the mean response time for the RRF being not significantly different from that for the Original condition. The response time in all the other algorithm conditions was significantly different from the response time in the in Original condition. The response time for the RRF algorithm was not significantly different from the multiscale Retinex algorithm and was significantly different from all the other algorithm conditions. In terms of percent error, the Original and RRF conditions were not significantly different from each other. Both were significantly different from the other algorithm conditions in percent error. Of additional significance is the fact that one algorithm, POSHE, did worse than the Degraded condition, resulting in both longer response times and a higher percent error. While this may seem surprising, the POSHE algorithm creates large blocking effects that were still obvious in the image even after the blocking effect reduction filter was applied. The blocking effects could be seen as extra noise in the image which could result in slower search times and larger percent error, as the observers might interpret the noise of the blocking effects to be the signal from the target.

Figure 7 summarizes the results of the paired comparisons with rotation angles sorted by increasing response time. As can be seen, there were significant differences in both response time and percent error among the different rotation angles. This may have occurred due to the differences in terrain density in different areas of the terrain board. While the terrain was fairly similar in content in the various areas of the terrain board, there were a few scenes (angles) in which the foliage was fairly sparse. This might result in a faster and more accurate search, as there would be fewer distracting elements that the observers would have to search through to find the target. Additionally, there were several scenes in which there was more dense foliage than in others and this would tend to slow search times and make them less accurate, as there would be more distracting elements to confuse the target with.

Figure 8 summarizes the results of the paired comparisons with camera tilt. The figure shows that there was no significant difference among the camera tilts for either response time or percent error. This is not surprising, as the scene would not change drastically as the camera was tilted. The density of the foliage in the scene appears to be a critical element in terms of response time and percent error and, for the most part, the overall density of the foliage in the scene would remain the same as the tilt was changed.

Figure 9 summarizes the results of the paired comparisons with quadrant sorted by increasing response time in the upper panel and increasing percent error in the lower panel. As the figure shows, there was a significant difference among the quadrants in terms of both response time and percent error. The fastest response times were in the upper right quadrant, followed by the upper left, lower left and lower right quadrant. This may have occurred due to a particular search strategy utilized by the observers; observers revealed that the first location they started to look in was the upper right quadrant and then followed a counterclockwise pattern, ending in the lower right quadrant. If they did not find the target in the initial pass through each quadrant, they would repeat the pattern, spending more time in each quadrant. This strategy was utilized by the majority of the observers, regardless of the fact that they were not prompted to adopt any particular search strategy.

The objective of this work was to use the assessment methodology developed in Neriani et al.[4] to assess the degree of contrast enhancement provided by six different algorithms. Based on the results discussed above, it would appear that both the RRF algorithm and (perhaps) the multiscale Retinex algorithm are promising algorithms in terms of contrast enhancement. The RRF algorithm condition had both a mean log response time and a mean log percent error that were not significantly different from the Original condition and the multiscale Retinex algorithm condition had a mean log response time that was not significantly different from the RRF algorithm condition. The other four algorithms (Autolevels, Global Histogram Equalization, BBFHE, and POSHE) did not perform to the same level as the RRF and multiscale Retinex algorithms. In conclusion, we will further investigate the enhancing effects of both the RRF and multiscale Retinex algorithm by testing them using different stimuli (different target, different terrain, and different lighting), as we believe these two algorithms to provide the greatest benefit to our research program.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the excellent support provided by Sheldon E. Unger and David W. Sivert, who captured the images, and by Martha A. Hausmann, who collected the data, all of General Dynamics-Advanced Information Engineering Systems, Dayton, OH.

REFERENCES

1. S. Marsi, G. Ramponi, and S. Carrato, "Image Contrast Enhancement using a Recursive Rational Filter", *IEEE IST: International Workshop on Imaging Systems and Techniques*, 29-34 (May 2004).
2. Z. Rahman, D. Jobson, and G. Woodell, "Retinex processing for automatic image enhancement", *Journal of Electronic Imaging*, **13**(1), 100-110 (January 2004).
3. D. Sale, R. Schultz, and R. Szczerba, "Super-Resolution Enhancement of Night Vision Image Sequences", *IEEE International Conference on Systems, Man, and Cybernetics*, **3**, 8-11 (October 2000).
4. K.E. Neriani, T.J. Herbranson, A.R. Pinkus, C.M. Task, H.L. Task, "Visual performance-based enhancement methodology: an investigation of three Retinex algorithms." in *Enhanced and Synthetic Vision,* J.G. Verly, ed., *Proc. SPIE 5802,* (March 2005).
5. E. Land, "The retinex", *American Scientist*, **52**, 247-264 (1964).
6. E. Land and J. McCann, "Lightness and Retinex theory", *Journal of the Optical Society of America*, **61**, 1-11 (1971).
7. J-Y. Kim, L-S. Kim, and S-H. Hwang, "An advanced contrast enhancement using partially overlapped sub-block histogram equalization", *IEEE Transactions on Circuits and Systems for Video Technology*, **11**(4), 475- 484 (April 2001).
8. F. Lamberti, B. Montrucchio, and A. Sanna, "BBFHE: Block-based binomial filtering histogram equalization", *WSEAS Transaction on Information Science and Applications*, **1**(6), 1591-1596, 2004.